Maximizing Java Performance on Mac OS X

Session 625

Maximizing Java Performance on Mac OS X

Jim Laskey, Victor Hernandez, and Gerard Ziemski
Java Runtime Technologies and Platform Classes
jlaskey, vhernandez, gziemski@apple.com

Introduction

- Understand performance of your Java application
- Runtime improvements
  - HotSpot optimized for G5
- Performance opportunities in Java 1.4.1
  - Aggressive inlining
  - New I/O
- Graphics performance

Maximizing Java Performance

Java VM Changes

Jim Laskey
Code Generation and Runtimes
jlaskey@apple.com

Overview

- Introduction to the G5
- Performance comparison G4 vs. G5
- Changes to the HotSpot Interpreter and JITc
- Changes to the HotSpot Runtime

Introduction to the G5

- G5 and Java developers
  - Faster CPU, bus, and floating-point
- G5 and the Java VM
  - VM uses G5 instruction set
  - 64-bit wins without the penalty of 64-bit addresses
- G5 and your Java application

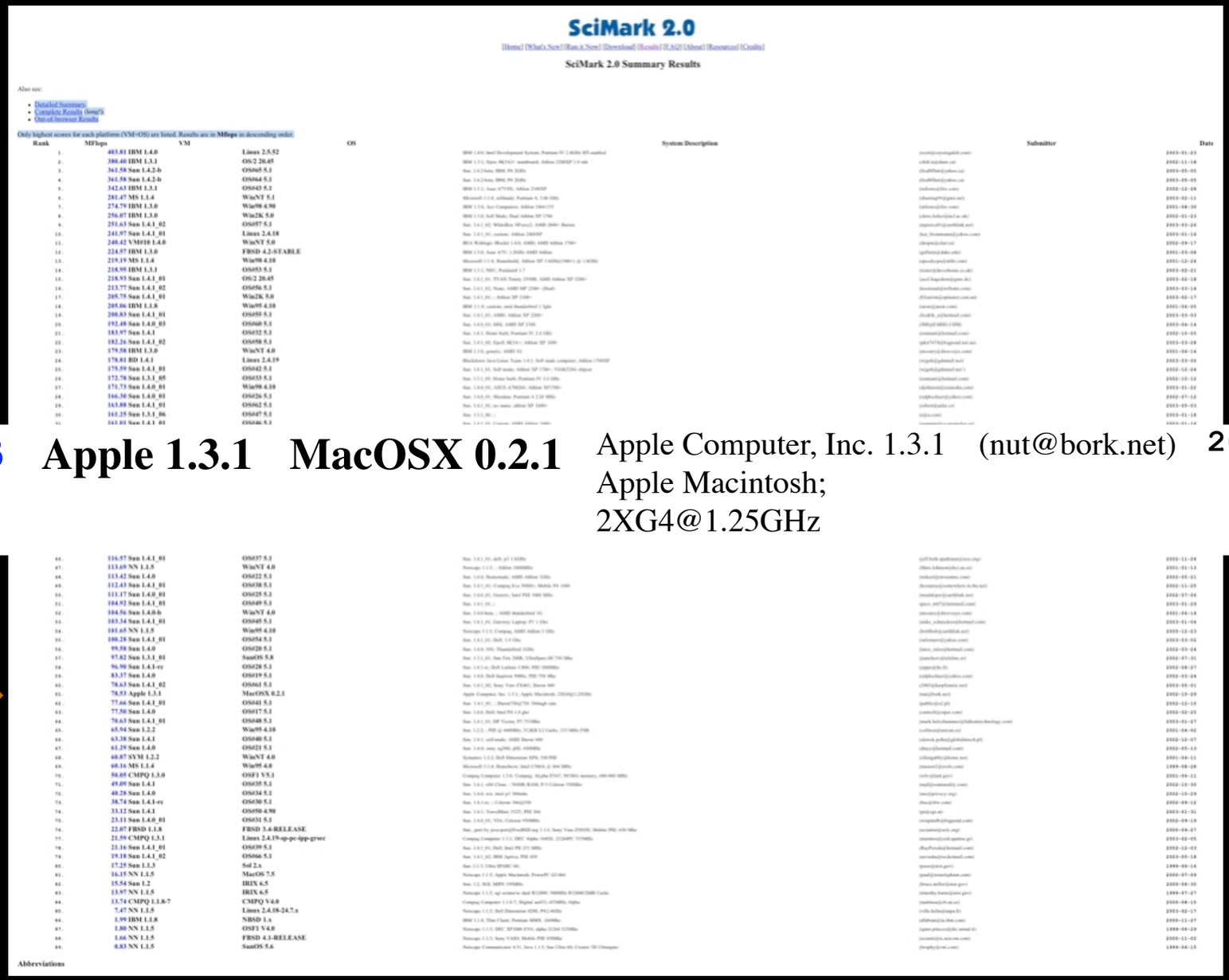
Performance Comparison G4 vs. G5

- Benchmarks
  - Internally we use SpecJVM, SciMark, Volano, and Caffeine
  - SpecJVM is best, but fair use requires public posting
  - SciMark is a decent benchmark with a focus on scientific/ engineering computation

Performance Comparison G4 vs. G5

SciMark 2.0

- http://math.nist.gov/cgi-bin/ScimarkSummary



SciMark 2.0

[Home] [What's New] [Run it Now] [Download] [Results] [FAQ] [About] [Resources] [Credits]

SciMark 2.0 Summary Results

Also see:

- Detailed Summary
- Complete Results (long!)
- Out-of-browser Results

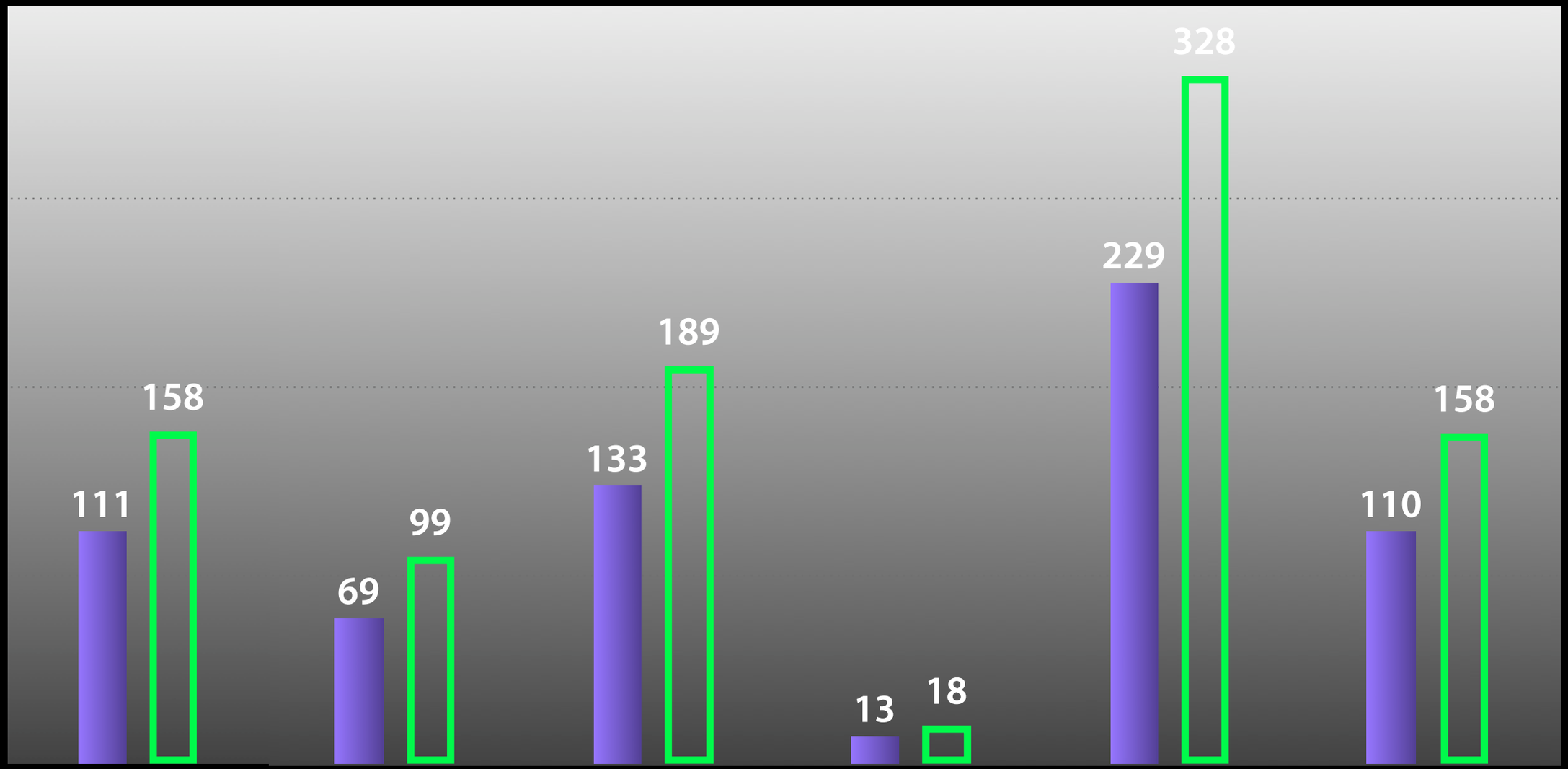
Only highest scores for each platform (VM+OS) are listed. Results are in **Mflops** in descending order.

Rank	Mflops	VM	OS	System Description	Submitter	Date
1.	403.81	IBM 1.4.0	Linux 2.5.52	IBM 1.4.0, Intel Development System, Pentium IV 2.1GHz HT-enabled	(scott@corporatejerk.com)	2003-01-23
2.	380.40	IBM 1.3.1	OS/2 20.45	IBM 1.3.1, Spec 8KHA2+ mainboard, Athlon 2100XP 1.8 mh	(chrh.ca@shaw.ca)	2002-11-14
3.	361.58	Sun 1.4.2-b	OS/66 5.1	Sun, 1.4.2-beta, IBM, P4 2GHz	(JoeIBMJoe@yahoo.ca)	2003-05-05
4.	361.58	Sun 1.4.2-b	OS/66 5.1	Sun, 1.4.2-beta, IBM, P4 2GHz	(JoeIBMJoe@yahoo.ca)	2003-05-05
5.	342.63	IBM 1.3.1	OS/43 5.1	IBM 1.3.1, Asus A7V8X, Athlon 2100XP	(nehrenc@fsu.com)	2002-12-28
6.	283.47	MS 1.1.4	WinNT 5.1	Microsoft 1.1.4, softmade, Pentium 4, 3.06 GHz	(cheating99@juno.net)	2003-02-13
7.	274.79	IBM 1.3.0	Win98 4.90	IBM 1.3.0, Ace Computers, Athlon 1466/333	(nehrenc@fsu.com)	2001-08-30
8.	256.07	IBM 1.3.0	Win2K 5.0	IBM 1.3.0, Self Made, Dual Athlon XP 1700	(chris.forkel@ncl.ac.uk)	2002-01-23
9.	251.63	Sun 1.4.1_02	OS/057 5.1	Sun, 1.4.1_02, WhiteBox 'nForce2', AMD 2000+ Barton	(mpiercel@yorkhtek.net)	2003-03-26
10.	241.97	Sun 1.4.1_01	Linux 2.4.18	Sun, 1.4.1_01, custom, Athlon 2400XP	(kai_brantmann@yhoo.com)	2003-01-14
11.	240.42	VM#10 1.4.0	WinNT 5.0	IBUK Weltlogic iBuckit 1.4.0, AMD, AMD Athlon 1700+	(dreppin@chat.ru)	2002-09-17
12.	224.57	IBM 1.3.0	FBSD 4.2-STABLE	IBM 1.3.0, Asus A7V, 1.2GHz AMD Athlon	(gollatin@duke.edu)	2001-03-08
13.	219.19	MS 1.1.4	Win98 4.10	Microsoft 1.1.4, Homebuild, Athlon XP 1.6GHz(1900+) @ 1.8GHz	(speedycpu@attbi.com)	2001-12-24
14.	218.95	IBM 1.3.1	OS/053 5.1	IBM 1.3.1, NEC, Pentium4 1.7	(tester@devonline.co.uk)	2003-02-21
15.	218.93	Sun 1.4.1_01	OS/2 20.45	Sun, 1.4.1_01, TT-ATi Trinity 256MB, AMD Athlon XP 2200+	(mcf.bagshaw@gmx.de)	2003-02-18
16.	213.77	Sun 1.4.1_02	OS/056 5.1	Sun, 1.4.1_02, Note, AMD MP 2200+ (Dual)	(fcurran@softhus.com)	2003-03-14
17.	205.75	Sun 1.4.1_01	Win2K 5.0	Sun, 1.4.1_01, -, Athlon XP 2100+	(Curran@optimizer.com.au)	2003-02-17
18.	205.06	IBM 1.3.0	Win95 4.10	IBM 1.3.0, custom, and thunderbird 3.5ghz	(anon@anon.com)	2001-04-05
19.	200.83	Sun 1.4.1_01	OS/058 5.1	Sun, 1.4.1_01, AMD, Athlon XP 2200+	(fredrik_s@hotmail.com)	2003-03-03
20.	192.48	Sun 1.4.0_03	OS/060 5.1	Sun, 1.4.0_03, MSI, AMD XP 2100	(Mdyd-MSI@163.com)	2003-04-14
21.	183.97	Sun 1.4.1	OS/032 5.1	Sun, 1.4.1, Home built, Pentium IV 2.4 GHz	(vonkunt@hotmail.com)	2003-03-05
22.	182.26	Sun 1.4.1_02	OS/058 5.1	Sun, 1.4.1_02, EpoX 8KHA+, Athlon XP 1600	(pkn7474@biggend.net.au)	2003-03-28
23.	179.58	IBM 1.3.0	WinNT 4.0	IBM 1.3.0, generic, AMD 1G	(mcrary@dreezys.com)	2001-04-14
24.	178.81	BD 1.4.1	Linux 2.4.19	Blackdown Java-Linux Team 1.4.1, Self made computer, Athlon 1700XP	(vigabl@gdnmail.net)	2003-03-04
25.	175.59	Sun 1.4.1_01	OS/042 5.1	Sun, 1.4.1_01, Self made, Athlon XP 1700+, 512MCT266 chipset	(vigabl@gdnmail.net')	2002-12-24
26.	172.78	Sun 1.3.1_05	OS/033 5.1	Sun, 1.3.1_05, Home built, Pentium IV 2.4 GHz	(vonkunt@hotmail.com)	2002-10-12
27.	171.73	Sun 1.4.0_01	Win98 4.10	Sun, 1.4.0_01, ASUS A7M266, Athlon XP1700+	(dpdineen@centrecbs.com)	2003-01-22
28.	166.30	Sun 1.4.0_01	OS/026 5.1	Sun, 1.4.0_01, Mesidite, Pentium 4 2.26 MHz	(ralphschuet@yahoo.com)	2002-07-10
29.	163.88	Sun 1.4.1_01	OS/062 5.1	Sun, 1.4.1_01, no name, athlon XP 1600+	(robert@safai.ca)	2003-03-03
30.	161.25	Sun 1.3.1_06	OS/047 5.1	Sun, 1.3.1_06, -	(s@k.com)	2003-01-18
31.	161.01	Sun 1.4.1_01	OS/046 5.1	Sun, 1.4.1_01, Custom, AMD Athlon 1600+	(vonkunt@hotmail.com)	2003-01-14
61.	**78.53**	**Apple 1.3.1**	**MacOSX 0.2.1**	Apple Computer, Inc. 1.3.1 Apple Macintosh; 2XG4@1.25GHz	(nut@bork.net)	2002-10-29
62.	116.57	Sun 1.4.1_01	OS/037 5.1	Sun, 1.4.1_01, dell, p3 1.4GHz	(jeff.bork.sparkman@xxx.org)	2002-11-24
63.	113.69	NN 1.1.5	WinNT 4.0	Netscape 1.1.5, -, Athlon 1800MHz	(Marc.Johnson@ci.on.us)	2001-01-13
64.	113.42	Sun 1.4.0	OS/022 5.1	Sun, 1.4.0, Homemade, AMD Athlon 1GHz	(mikael@devonline.com)	2002-05-21
65.	112.43	Sun 1.4.1_01	OS/038 5.1	Sun, 1.4.1_01, Compaq Evo N800c, Mobile P4 1600	(hcouture@somewhere.in.the.net)	2002-11-05
66.	111.17	Sun 1.4.0_01	OS/025 5.1	Sun, 1.4.0_01, Generic, Intel PIII 1000 MHz	(mudshiper@earthlink.net)	2002-07-04
67.	104.92	Sun 1.4.1_01	OS/049 5.1	Sun, 1.4.1_01, -	(poco_m07@hotmail.com)	2003-01-29
68.	104.56	Sun 1.4.0-b	WinNT 4.0	Sun, 1.4.0-beta, -, AMD thunderbird 1G	(mcrary@dreezys.com)	2001-04-14
69.	103.34	Sun 1.4.1_01	OS/045 5.1	Sun, 1.4.1_01, Gateway Laptop, P1 1 Ghz	(mike_schneider@hotmail.com)	2003-01-04
70.	101.65	NN 1.1.5	Win95 4.10	Netscape 1.1.5, Compaq, AMD Athlon 1 GHz	(bottlbob@earthlink.net)	2003-12-23
71.	100.28	Sun 1.4.1_01	OS/054 5.1	Sun, 1.4.1_01, Dell, 3.9 Ghz	(nehrenc@yahoo.com)	2003-03-02
72.	99.58	Sun 1.4.0	OS/020 5.1	Sun, 1.4.0, NN, Thunderbird 1GHz	(latex_ruler@hotmail.com)	2002-03-24
73.	97.82	Sun 1.3.1_01	SunOS 5.8	Sun, 1.3.1_01, Sun Ftw 360R, UltraSparc-II 750 Mhz	(jsanchez@sdsbbs.ca)	2002-07-31
74.	96.90	Sun 1.4.1-rc	OS/028 5.1	Sun, 1.4.1-rc, Dell Latitude C600, PIII 1000Mhz	(cpgt@ikc.fi)	2002-08-27
75.	83.37	Sun 1.4.0	OS/019 5.1	Sun, 1.4.0, Dell Inspiron 5000e, PIII 750 Mhz	(ralphschuet@yahoo.com)	2002-03-24
76.	78.63	Sun 1.4.1_02	OS/061 5.1	Sun, 1.4.1_02, Sony Vaio FX401, Duron 800	(2005@keepforever.net)	2003-05-01
77.	78.53	Apple 1.3.1	MacOSX 0.2.1	Apple Computer, Inc. 1.3.1, Apple Macintosh, 2XG4@1.25GHz	(nut@bork.net)	2002-10-29
78.	77.66	Sun 1.4.1_01	OS/040 5.1	Sun, 1.4.1_01, -, Duron750@750 366mgb ram	(pablo@cpl.ch)	2002-12-10
79.	77.50	Sun 1.4.0	OS/017 5.1	Sun, 1.4.0, Dell, Intel P4 1.4 ghz	(cantor@caspar.com)	2002-02-25
80.	70.63	Sun 1.4.1_01	OS/048 5.1	Sun, 1.4.1_01, HP Vectra, P3 733Mhz	(mark.kolhammer@fabmatics.hecksgy.com)	2003-01-27
81.	65.94	Sun 1.2.2	Win95 4.10	Sun, 1.2.2, -, PIII @ 666MHz, 512KB L2 Cache, 133 MHz FSB	(collins@unicon.ca)	2001-04-02
82.	63.38	Sun 1.4.1	OS/040 5.1	Sun, 1.4.1, self-made, AMD Duron 800	(derek.pelkie@globalmech.pf)	2002-12-07
83.	61.29	Sun 1.4.0	OS/021 5.1	Sun, 1.4.0, sony xg500, p6H, 450MHz	(dhyo@hotmail.com)	2002-05-13
84.	60.87	SVM 1.2.2	WinNT 4.0	Symantec 1.2.2, Dell Dimension XPS, 550 PIII	(vfrag@fly.cybertec.net)	2001-04-11
85.	60.16	MS 1.1.4	Win95 4.0	Microsoft 1.1.4, Homebrew, Intel C700A @ 466 MHz	(mason2@corth.com)	1999-08-28
86.	50.05	CMPQ 1.3.0	OSF1 V5.1	Compaq Computer 1.3.0, Compaq, Alpha EV67, 90 MA memory, 600-900 MHz	(wlv@dnf.gov)	2001-05-11
87.	49.09	Sun 1.4.1	OS/035 5.1	Sun, 1.4.1, x86 Clone - 384Mb RAM, P-3 Celeron 550Mhz	(mpl@commodity.com)	2002-10-30
88.	40.28	Sun 1.4.0	OS/034 5.1	Sun, 1.4.0, x/c, intel p3 700mhz	(mc@privacy.org)	2002-10-29
89.	38.74	Sun 1.4.1-rc	OS/030 5.1	Sun, 1.4.1-rc, -, Celeron 366@550	(bsc@ibx.com)	2002-09-12
90.	33.12	Sun 1.4.1	OS/050 4.90	Sun, 1.4.1, TransMeta 5535, PIII 366	(pcl@cgs.nl)	2003-01-31
91.	23.11	Sun 1.4.0_01	OS/031 5.1	Sun, 1.4.0_01, VIA, Celeron 950MHz	(wiapatch@biggend.com)	2002-09-19
92.	22.07	FBSD 1.1.8	FBSD 3.4-RELEASE	Sun, port by java-port@FreeBSD.org 1.1.8, Sony Vaio Z505JS, Mobile P3E, 650 Mhz	(scranton@ucls.org)	2000-09-27
93.	21.59	CMPQ 1.3.1	Linux 2.4.19-sp-pc-ipp-grsec	Compaq Computer 1.3.1, DEC Alpha 164SX, 21264PC 550MHz	(martinez@cold.spartus.gr)	2003-02-05
94.	21.16	Sun 1.4.1_01	OS/039 5.1	Sun, 1.4.1_01, Dell, Intel PII 251 MHz	(RayPowder@hotmail.com)	2002-12-03
95.	19.18	Sun 1.4.1_02	OS/066 5.1	Sun, 1.4.1_02, IBM Aptiva, PIII 450	(mrcrabe@cox.hotmail.com)	2003-05-14
96.	17.25	Sun 1.3.3	Sol 2.x	Sun, 1.3.3, Ultra SPARC 60,	(jmast@mit.gov)	1999-04-14
97.	16.15	NN 1.1.5	MacOS 7.5	Netscape 1.1.5, Apple Macintosh, PowerPC G3 466	(paul@roanokphant.com)	2000-07-04
98.	15.54	Sun 1.2	IRIX 6.5	Sun, 1.2, SGI, MIPS 195MHz	(bruce.miller@mit.gov)	2000-04-30
99.	13.97	NN 1.1.5	IRIX 6.5	Netscape 1.1.5, sgi octane/oe dual R12000, 300MHz R12000/2MB Cache	(timothy.hume@mit.gov)	1999-07-27
100.	13.74	CMPQ 1.1.8-7	CMPQ V4.0	Compaq Computer 1.1.8-7, Digital au4351, 433MHz Alpha	(matthias@cfr.de.us)	2000-04-15
101.	7.47	NN 1.1.5	Linux 2.4.18-24.7.x	Netscape 1.1.5, Dell Dimension X295, P4/2.4GHz	(villa.helm@mapo.fr)	2003-02-17
102.	1.99	IBM 1.1.8	NBSD 1.x	IBM 1.1.8, Thin Client, Pentium MMX -266Mhz	(dfabian@in.ibm.com)	2000-11-07
103.	1.80	NN 1.1.5	OSF1 V4.0	Netscape 1.1.5, DEC XP1000 EV6, alpha 21264 525Mhz	(john.prisco@dec.nmsd.it)	1999-04-29
104.	1.66	NN 1.1.5	FBSD 4.1-RELEASE	Netscape 1.1.5, Sony VAIO, Mobile PIII 450Mhz	(scranit@ix.netcom.com)	2000-12-02
105.	0.83	NN 1.1.5	SunOS 5.6	Netscape Communicator 4.51, Java 1.1.5, Sun Ultra 60, Creator 3D Elite	(bopby@vnt.com)	1999-04-15

Abbreviations

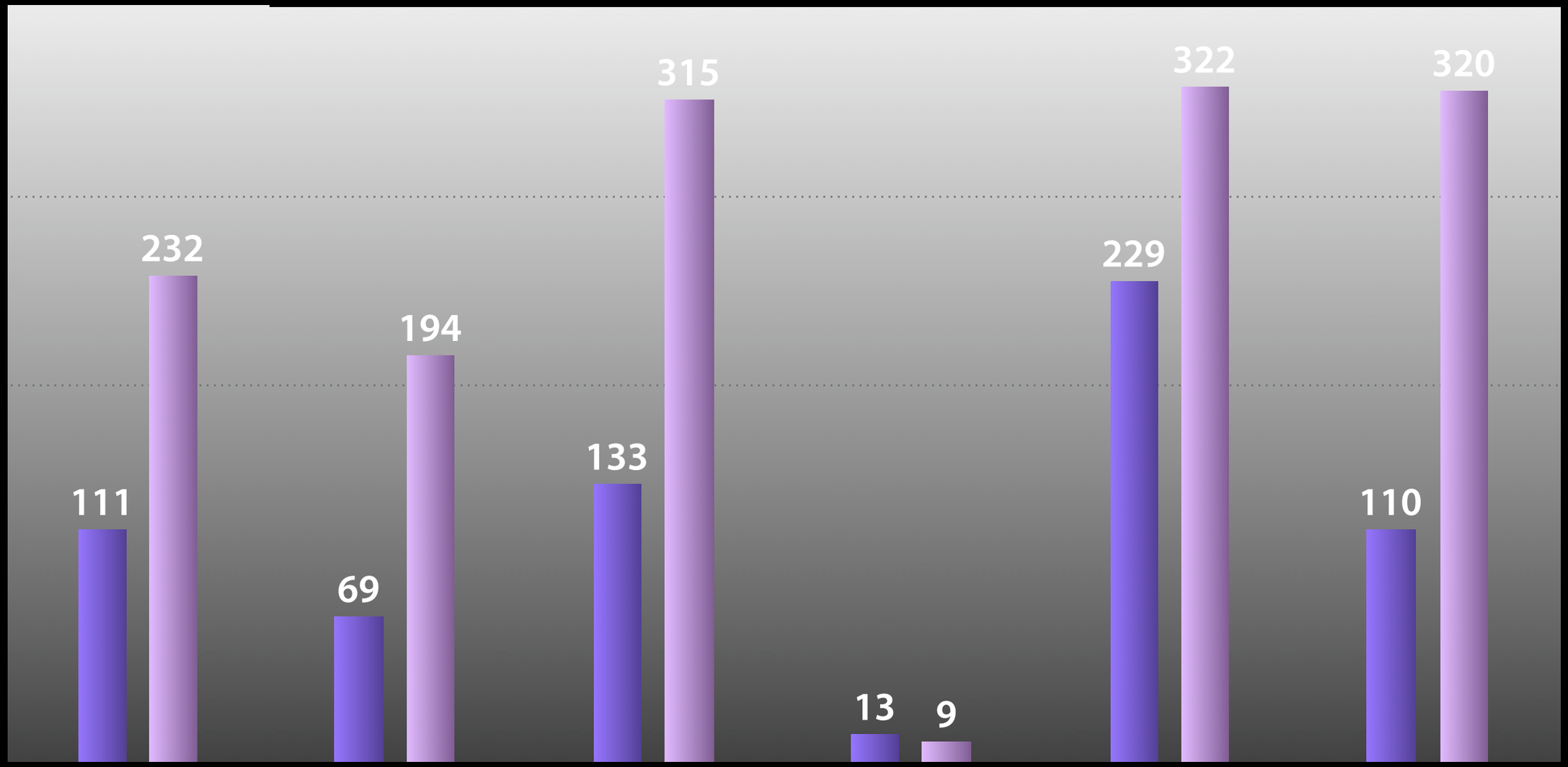
Performance Comparison G4 vs. G5

SciMark on HotSpot Client



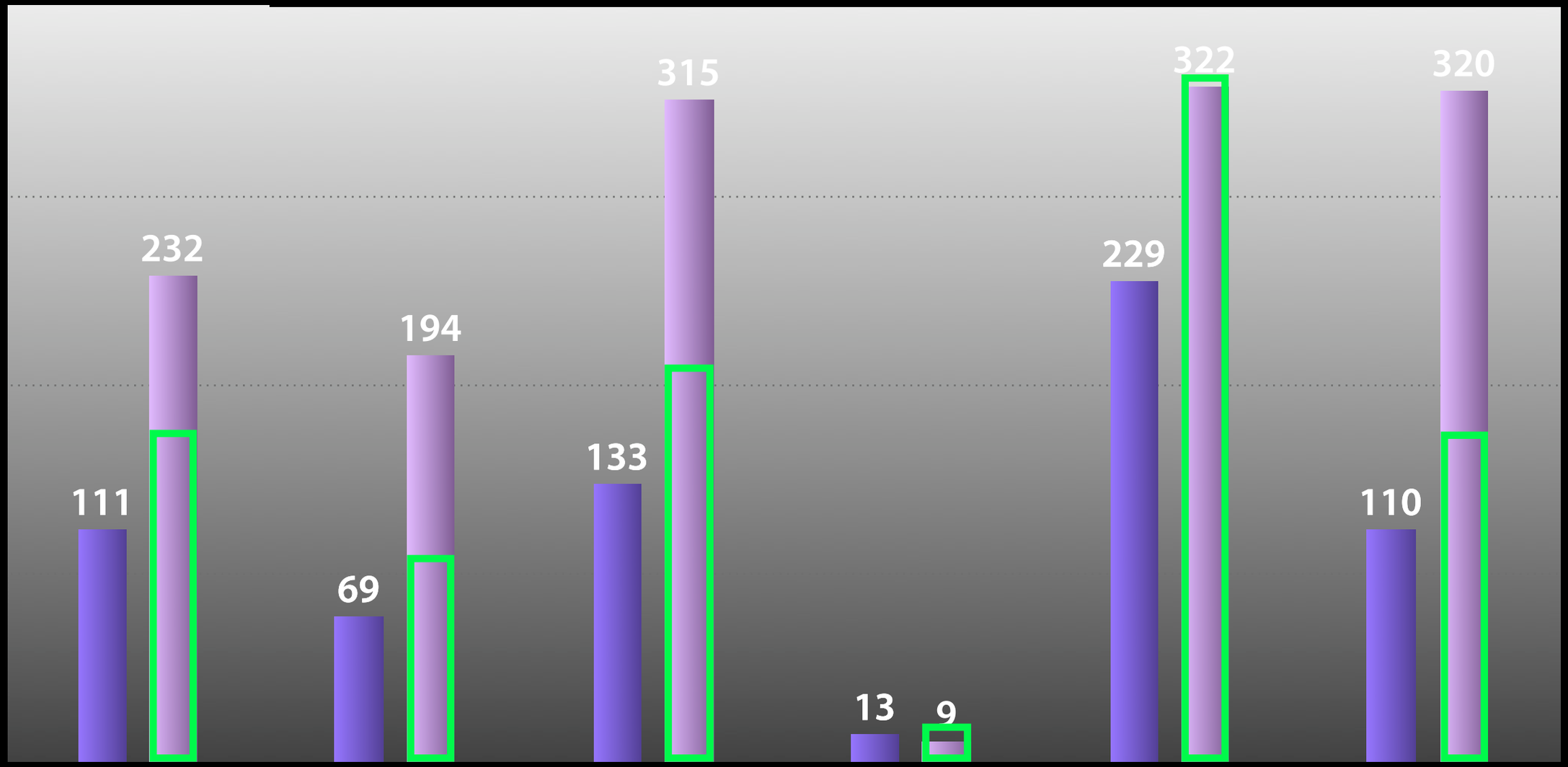
Performance Comparison G4 vs. G5

SciMark on HotSpot Client



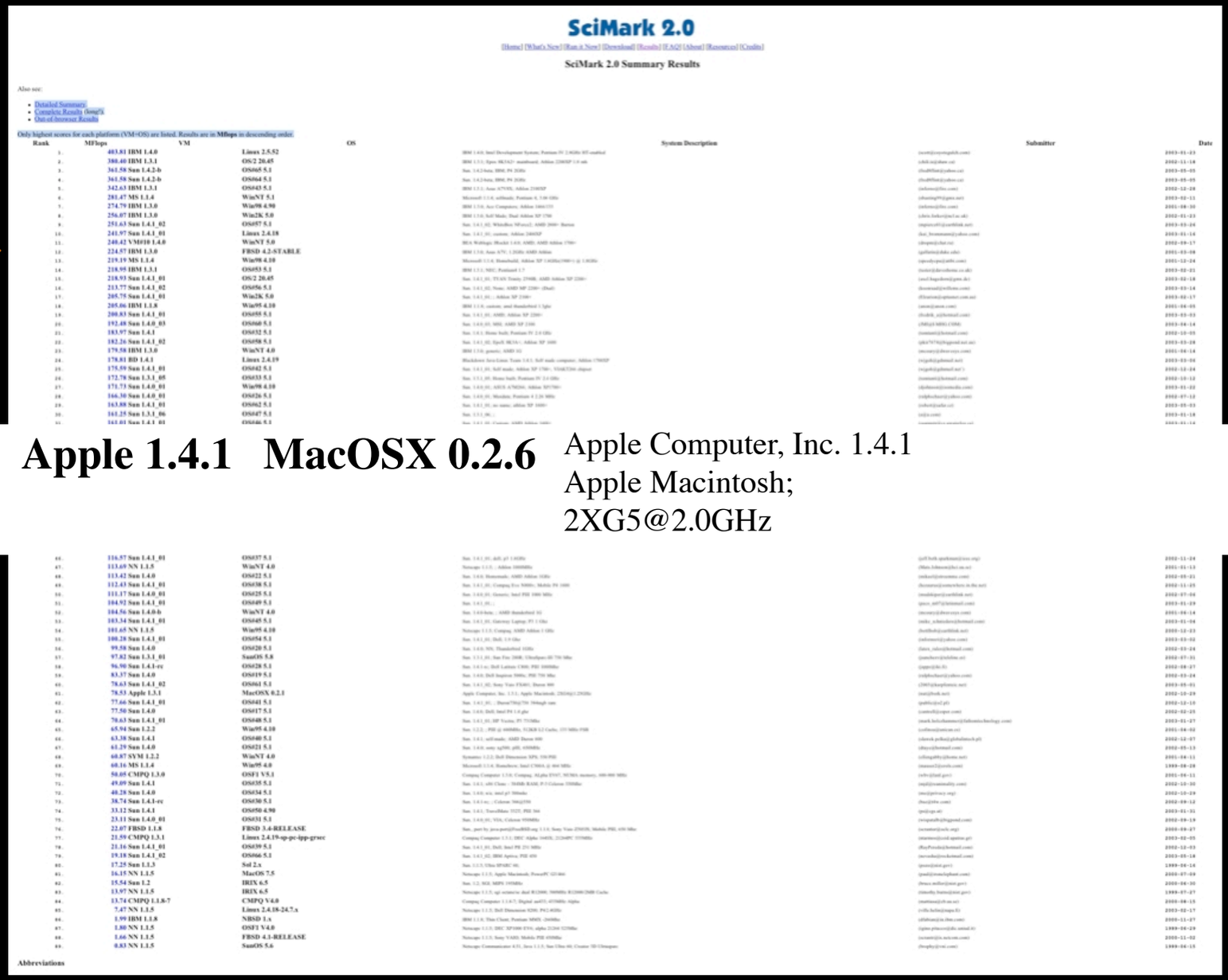
Performance Comparison G4 vs. G5

SciMark on HotSpot Client



Performance Comparison G4 vs. G5

SciMark 2.0



SciMark 2.0

[Home] [What's New] [Run it Now] [Download] [Results] [FAQ] [About] [Resources] [Credits]

SciMark 2.0 Summary Results

Also see:

- Detailed Summary
- Complete Results (long!)
- Out-of-browser Results

Only highest scores for each platform (VM+OS) are listed. Results are in **Mflops** in descending order.

Rank	Mflops	VM	OS	System Description	Submitter	Date
1.	403.81	IBM 1.4.0	Linux 2.5.52	IBM 1.4.0, Intel Development System, Pentium IV 2.0GHz HT-enabled	(scott@corporatejerk.com)	2003-01-23
2.	380.40	IBM 1.3.1	OS/2 20.45	IBM 1.3.1, Epox 8KTA2+ motherboard, Athlon 2100XP 1.8 mh	(chrh.ca@shaw.ca)	2002-11-14
3.	361.58	Sun 1.4.2-b	OS/664 5.1	Sun, 1.4.2-beta, IBM, P4 2GHz	(DonBNunn@yahoo.ca)	2003-05-05
4.	361.58	Sun 1.4.2-b	OS/664 5.1	Sun, 1.4.2-beta, IBM, P4 2GHz	(DonBNunn@yahoo.ca)	2003-05-05
5.	342.63	IBM 1.3.1	OS/643 5.1	IBM 1.3.1, Asus A7V8X, Athlon 2100XP	(nelsonw@fsu.com)	2002-12-28
6.	283.47	MS 1.1.4	WinNT 5.1	Microsoft 1.1.4, softmade, Pentium 4, 3.06 GHz	(shanting99@gmx.net)	2003-02-13
7.	274.79	IBM 1.3.0	Win98 4.90	IBM 1.3.0, Acc Computers, Athlon 1466/133	(nelsonw@fsu.com)	2001-08-30
8.	256.07	IBM 1.3.0	Win2K 5.0	IBM 1.3.0, Self Made, Dual Athlon XP 1700	(chris.forkel@ncl.ac.uk)	2002-01-23
9.	251.63	Sun 1.4.1_02	OS/057 5.1	Sun, 1.4.1_02, WhiteBox 9Force22, AMD 2800+ Barton	(mpiercy01@earthlink.net)	2003-03-26
10.	241.97	Sun 1.4.1_01	Linux 2.4.18	Sun, 1.4.1_01, custom, Athlon 2400XP	(kai_brantmann@yhoo.com)	2003-01-14
11.	240.42	VM#10 1.4.0	WinNT 5.0	IBA Wellspin iRocket 1.4.0, AMD, AMD Athlon 1700+	(dwpate@chat.ru)	2002-09-17
12.	224.57	IBM 1.3.0	FBSD 4.2-STABLE	IBM 1.3.0, Asus A7V, 1.2GHz AMD Athlon	(gollatta@duke.edu)	2001-03-08
13.	219.19	MS 1.1.4	Win98 4.10	Microsoft 1.1.4, Homebuild, Athlon XP 1.6GHz(1900+) @ 1.8GHz	(spoudyan@attbi.com)	2001-12-24
14.	218.95	IBM 1.3.1	OS/053 5.1	IBM 1.3.1, NEC, Pentium4 1.7	(tester@devotime.co.uk)	2003-02-21
15.	218.93	Sun 1.4.1_01	OS/2 20.45	Sun, 1.4.1_01, TT-ATi Trinity 256MB, AMD Athlon XP 2200+	(mcf.hagedorn@gmx.de)	2003-02-18
16.	213.77	Sun 1.4.1_02	OS/056 5.1	Sun, 1.4.1_02, Note, AMD MP 2200+ (Dual)	(floatman@wilkes.com)	2003-03-14
17.	205.75	Sun 1.4.1_01	Win2K 5.0	Sun, 1.4.1_01, -, Athlon XP 2100+	(Floatman@optimizer.com.au)	2003-02-17
18.	205.06	IBM 1.3.0	Win95 4.10	IBM 1.3.0, custom, and thunderbird 3.5ghz	(anon@anon.com)	2001-04-05
19.	200.83	Sun 1.4.1_01	OS/058 5.1	Sun, 1.4.1_01, AMD, Athlon XP 2200+	(fredrik_s@hotmail.com)	2003-03-03
20.	192.48	Sun 1.4.0_03	OS/060 5.1	Sun, 1.4.0_03, MSI, AMD XP 2100	(Mdjs@ MSU.EDU)	2003-04-14
21.	183.97	Sun 1.4.1	OS/032 5.1	Sun, 1.4.1, Home built, Pentium IV 2.4 GHz	(rustant@hotmail.com)	2002-10-05
22.	182.26	Sun 1.4.1_02	OS/058 5.1	Sun, 1.4.1_02, EpoX 8KHA+, Athlon XP 1600	(pkn7474@biggend.net.au)	2003-03-28
23.	179.58	IBM 1.3.0	WinNT 4.0	IBM 1.3.0, generic, AMD 1G	(mcrary@dnsrsys.com)	2001-04-14
24.	178.81	BD 1.4.1	Linux 2.4.19	Blackdown Java-Linux Team 1.4.1, Self made computer, Athlon 1700XP	(vigabl@gdnmail.net)	2003-03-04
25.	175.59	Sun 1.4.1_01	OS/042 5.1	Sun, 1.4.1_01, Self made, Athlon XP 1700+, NVidia nVidia chipset	(vigabl@gdnmail.net')	2002-12-24
26.	172.78	Sun 1.3.1_05	OS/033 5.1	Sun, 1.3.1_05, Home built, Pentium IV 2.4 GHz	(rustant@hotmail.com)	2002-10-12
27.	171.73	Sun 1.4.0_01	Win98 4.10	Sun, 1.4.0_01, ASUS A7M266, Athlon XP1700+	(dpdineen@centrecbs.com)	2003-01-22
28.	166.30	Sun 1.4.0_01	OS/026 5.1	Sun, 1.4.0_01, Mesidm, Pentium 4 2.26 MHz	(ralphschuler@yahoo.com)	2002-07-10
29.	163.88	Sun 1.4.1_01	OS/062 5.1	Sun, 1.4.1_01, no name, athlon XP 1600+	(robert@safai.ca)	2003-03-03
30.	161.25	Sun 1.3.1_06	OS/047 5.1	Sun, 1.3.1_06, -	(s@k.com)	2003-01-18
31.	161.01	Sun 1.4.1_01	OS/046 5.1	Sun, 1.4.1_01, Custom, AMD Athlon 1600+	(Comments@scimarkjava.com)	2003-01-14
32.	231.84	Apple 1.4.1	MacOSX 0.2.6	Apple Computer, Inc. 1.4.1, Apple Macintosh, 2XG5@2.0GHz	(scott@corporatejerk.com)	2003-6-27
33.	116.57	Sun 1.4.1_01	OS/037 5.1	Sun, 1.4.1_01, dell, p3 1.4GHz	(jeff.bolt.sparkman@xxx.org)	2002-11-24
34.	113.69	NN 1.1.5	WinNT 4.0	Netscape 1.1.5, -, Athlon 1800MHz	(Marc.Johnson@ci.on.us)	2001-01-13
35.	113.42	Sun 1.4.0	OS/022 5.1	Sun, 1.4.0, Homemade, AMD Athlon 1GHz	(mikael@dvtcomm.com)	2002-05-21
36.	112.43	Sun 1.4.1_01	OS/038 5.1	Sun, 1.4.1_01, Compaq Evo N800c, Mobile P4 1600	(hcouture@somewhere.in.the.net)	2002-11-05
37.	111.17	Sun 1.4.0_01	OS/025 5.1	Sun, 1.4.0_01, Generic, Intel P3E 1000 MHz	(mudskipr@earthlink.net)	2002-07-04
38.	104.92	Sun 1.4.1_01	OS/049 5.1	Sun, 1.4.1_01, -	(pico_m07@hotmail.com)	2003-01-29
39.	104.56	Sun 1.4.0-b	WinNT 4.0	Sun, 1.4.0-beta, -, AMD thunderbird 1G	(mcrary@dnsrsys.com)	2001-04-14
40.	103.34	Sun 1.4.1_01	OS/045 5.1	Sun, 1.4.1_01, Gateway Laptop, P1 1 Ghz	(mike_schneider@hotmail.com)	2003-01-04
41.	101.65	NN 1.1.5	Win95 4.10	Netscape 1.1.5, Compaq, AMD Athlon 1 GHz	(bottlbob@earthlink.net)	2003-12-23
42.	100.28	Sun 1.4.1_01	OS/054 5.1	Sun, 1.4.1_01, Dell, 3.0 Ghz	(nfortner@yahoo.com)	2003-03-02
43.	99.58	Sun 1.4.0	OS/020 5.1	Sun, 1.4.0, NN, Thunderbird 1GHz	(latex_ruler@hotmail.com)	2002-03-24
44.	97.82	Sun 1.3.1_01	SunOS 5.8	Sun, 1.3.1_01, Sun Fire 280R, UltraSparc-III 750 Mhz	(jsanchez@solidtw.co)	2002-07-31
45.	96.90	Sun 1.4.1-rc	OS/028 5.1	Sun, 1.4.1-rc, Dell Latitude C800, P3E 1000Mhz	(cpp@dkr.fi)	2002-08-27
46.	83.37	Sun 1.4.0	OS/019 5.1	Sun, 1.4.0, Dell Inspiron 5000e, P3E 750 Mhz	(ralphschuler@yahoo.com)	2002-03-24
47.	78.63	Sun 1.4.1_02	OS/061 5.1	Sun, 1.4.1_02, Sony Vaio FX401, Duron 800	(2005@keepforever.net)	2003-05-01
48.	78.53	Apple 1.3.1	MacOSX 0.2.1	Apple Computer, Inc. 1.3.1, Apple Macintosh, 2XG4@1.25GHz	(mat@bolt.net)	2002-10-29
49.	77.66	Sun 1.4.1_01	OS/041 5.1	Sun, 1.4.1_01, -, Duron750@750 566mph rate	(pablo@cpl.ch)	2002-12-10
50.	77.50	Sun 1.4.0	OS/017 5.1	Sun, 1.4.0, Dell, Intel P4 1.4 ghz	(cotrell@capat.com)	2002-02-25
51.	70.63	Sun 1.4.1_01	OS/048 5.1	Sun, 1.4.1_01, HP Vectra, P1 733Mhz	(mark.hutchinson@fabmatrixtechnology.com)	2003-01-27
52.	65.94	Sun 1.2.2	Win95 4.10	Sun, 1.2.2, , P3E @ 666MHz, 512KB L2 Cache, 133 MHz FSB	(collins@unicon.ca)	2001-04-02
53.	63.38	Sun 1.4.1	OS/040 5.1	Sun, 1.4.1, self-made, AMD Duron 800	(derek.pulko@globalintech.pl)	2002-12-07
54.	61.29	Sun 1.4.0	OS/021 5.1	Sun, 1.4.0, sony xg500, p6H, 450MHz	(dhays@hotmail.com)	2002-05-13
55.	60.87	SYM 1.2.2	WinNT 4.0	Symantec 1.2.2, Dell Dimension XPS, 550 PIII	(cflaig@fly.cybertec.net)	2001-04-11
56.	60.16	MS 1.1.4	Win95 4.0	Microsoft 1.1.4, Homebrew, Intel C700A @ 466 MHz	(mason2@corrib.com)	1999-08-28
57.	50.05	CMPQ 1.3.0	OSF1 V5.1	Compaq Computer 1.3.0, Compaq, Alpha EV67, 667MA memory, 600-900 MHz	(wlv@dnf.gov)	2001-06-11
58.	49.09	Sun 1.4.1	OS/035 5.1	Sun, 1.4.1, x86 Clone - 384Mb RAM, P-3 Celeron 550Mhz	(mpl@commodity.com)	2002-10-30
59.	40.28	Sun 1.4.0	OS/034 5.1	Sun, 1.4.0, n/a, intel p3 700mhz	(mc@privacy.org)	2002-10-29
60.	38.74	Sun 1.4.1-rc	OS/030 5.1	Sun, 1.4.1-rc, - Celeron 366@550	(bsc@bbn.com)	2002-09-12
61.	33.12	Sun 1.4.1	OS/050 4.90	Sun, 1.4.1, TransMeta 5535, PIII 366	(pa@cgs.at)	2003-01-31
62.	23.11	Sun 1.4.0_01	OS/031 5.1	Sun, 1.4.0_01, VIA, Celeron 950MHz	(wiapatah@biggend.com)	2002-09-19
63.	22.07	FBSD 1.1.8	FBSD 3.4-RELEASE	Sun, part by java-port@FreeBSD.org 1.1.8, Sony Vaio Z505JS, Mobile P3E, 650 Mhz	(scranton@ucls.org)	2000-09-27
64.	21.59	CMPQ 1.3.1	Linux 2.4.19-sp-pc-ipp-grsec	Compaq Computer 1.3.1, DEC Alpha 164SX, 21264PC 575Mhz	(martinez@cool.upatras.gr)	2003-02-05
65.	21.16	Sun 1.4.1_01	OS/039 5.1	Sun, 1.4.1_01, Dell, Intel P4 231 MHz	(RayPowder@hotmail.com)	2002-12-03
66.	19.18	Sun 1.4.1_02	OS/066 5.1	Sun, 1.4.1_02, IBM Aptiva, P3E 450	(mrcrabb@cs.hotmail.com)	2003-05-14
67.	17.25	Sun 1.3.3	Sol 2.x	Sun, 1.3.3, Ultra SPARC 60,	(pasat@nist.gov)	1999-04-14
68.	16.15	NN 1.1.5	MacOS 7.5	Netscape 1.1.5, Apple Macintosh, PowerPC G3 466	(paul@roanokphant.com)	2000-07-04
69.	15.54	Sun 1.2	IRIX 6.5	Sun, 1.2, SGI, MIPS 195MHz	(bruce.miller@nist.gov)	2000-04-30
70.	13.97	NN 1.1.5	IRIX 6.5	Netscape 1.1.5, sgi octane/oe dual R12000, 300MHz R12000/2MB Cache	(timothy.hunter@nist.gov)	1999-07-27
71.	13.74	CMPQ 1.1.8-7	CMPQ V4.0	Compaq Computer 1.1.8-7, Digital au4351, 433MHz Alpha	(mattman@cfr.ab.ca)	2000-04-15
72.	7.47	NN 1.1.5	Linux 2.4.18-24.7.x	Netscape 1.1.5, Dell Dimension XPS, P4/2.4GHz	(villa.helm@mapo.fi)	2003-02-17
73.	1.99	IBM 1.1.8	NBSD 1.x	IBM 1.1.8, Thin Client, Pentium MMX -266Mhz	(dfabian@in.ibm.com)	2000-11-27
74.	1.80	NN 1.1.5	OSF1 V4.0	Netscape 1.1.5, DEC XP1000 EV6, alpha 21264 525Mhz	(john.prisco@dec.nmsd.it)	1999-04-29
75.	1.66	NN 1.1.5	FBSD 4.1-RELEASE	Netscape 1.1.5, Sony VAIO, Mobile P3E 450Mhz	(scranton@ix.netcom.com)	2000-12-02
76.	0.83	NN 1.1.5	SunOS 5.6	Netscape Communicator 4.51, Java 1.1.5, Sun Ultra 60, Creator 3D Elite	(bwpby@vsnl.com)	1999-04-15

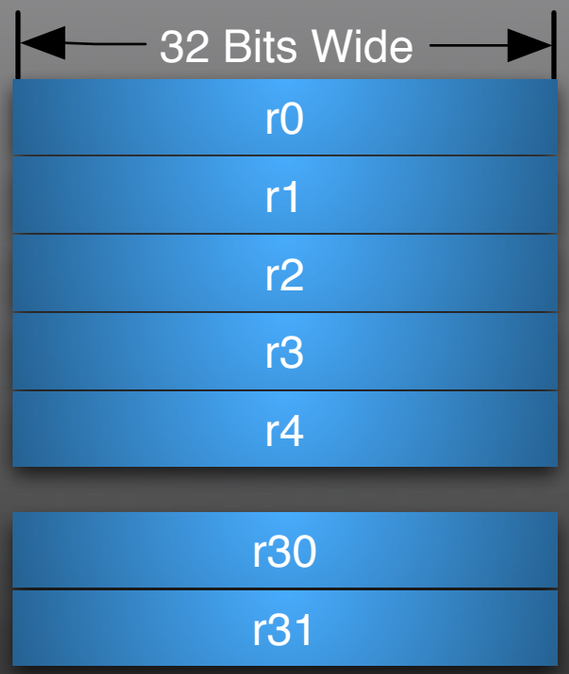
Abbreviations

Interpreter/JITc Changes

- Interpreter and JITc are enhanced with G5 instructions
- Java long support is improved using 64-bit instructions
- Float and double support is improved using new G5 floating point instructions

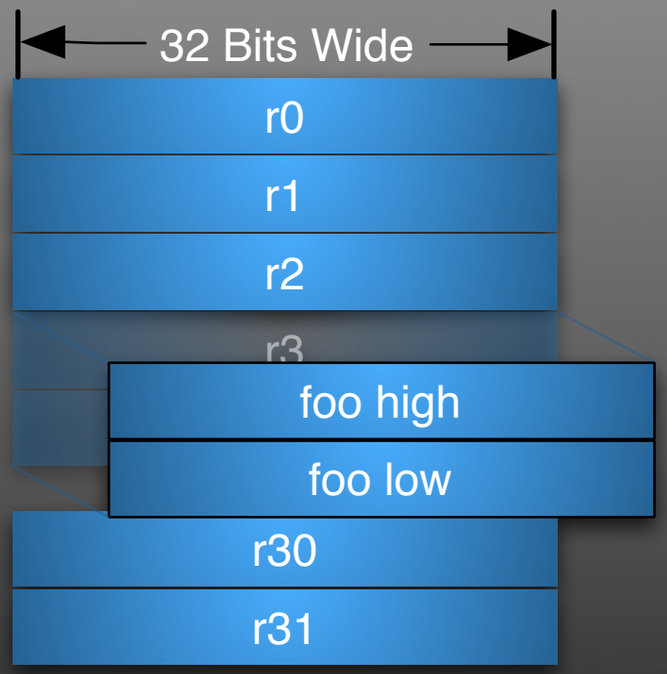
Interpreter/JITc Changes—Long

G4 General Purpose Registers



Interpreter/JITc Changes—Long

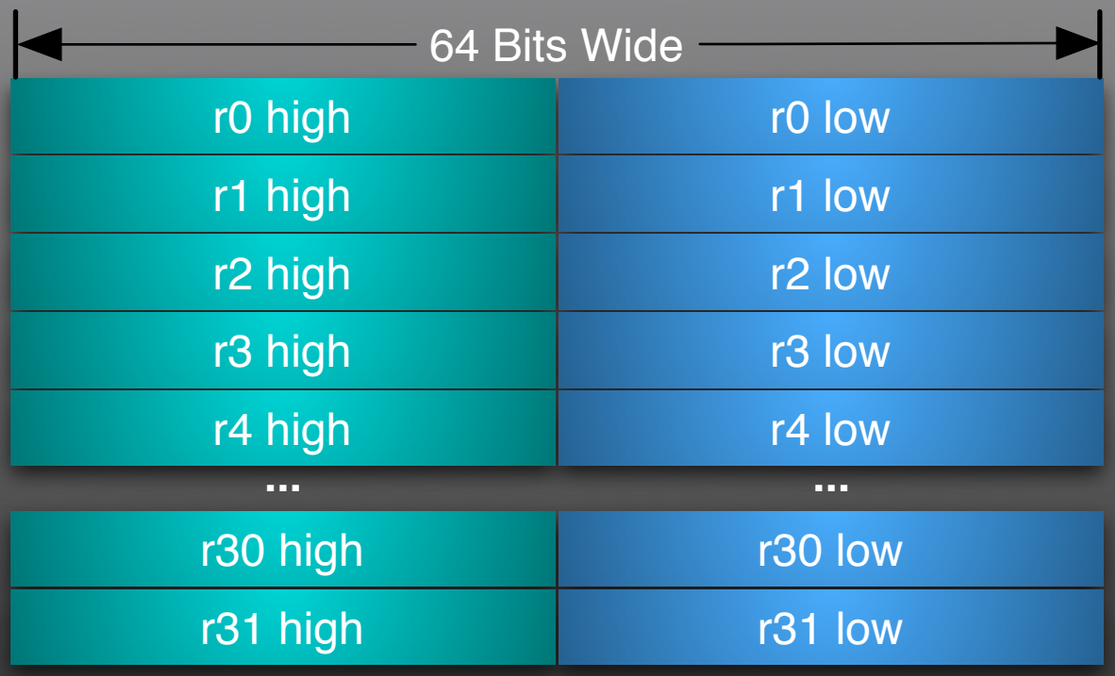
G4 General Purpose Registers




```
long foo;
foo = x + y;
```

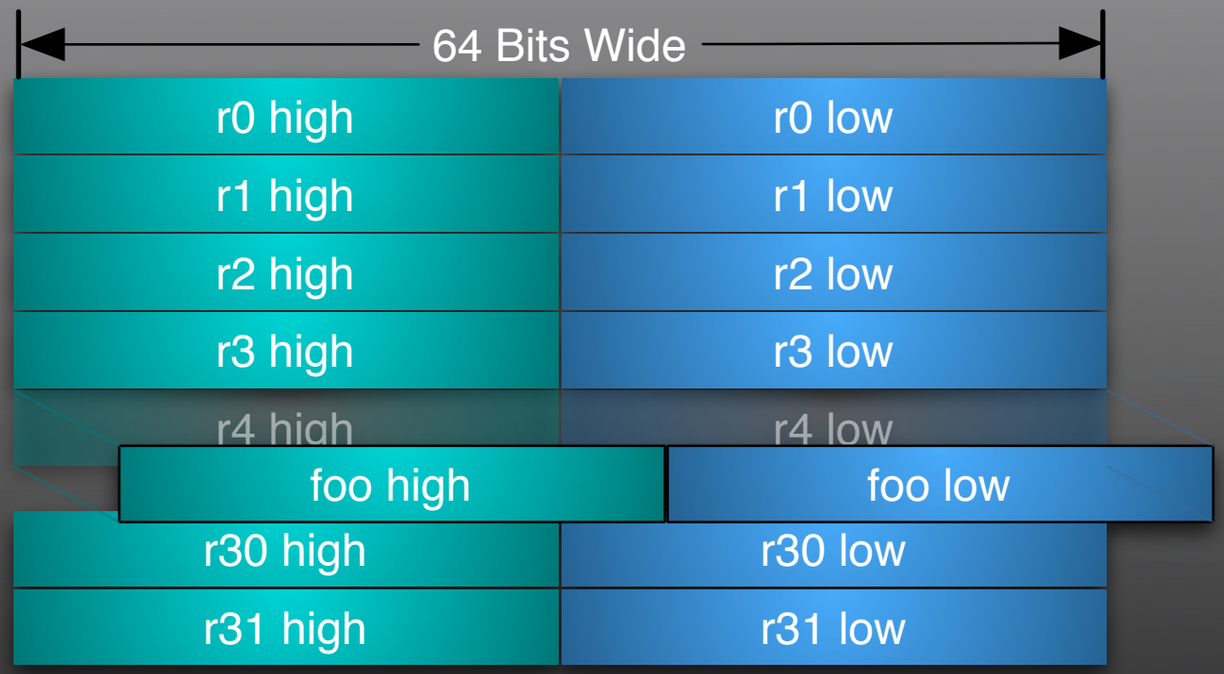
Interpreter/JITc Changes—Long

G5 General Purpose Registers



Interpreter/JITc Changes—Long

G5 General Purpose Registers




```
long foo;
foo = x + y;
```

Interpreter/JITc Changes—Long

- Single instruction loads and stores of longs

Interpreter/JITc Changes—Long

- Single instruction loads and stores of longs

```
long x = y;
```

Interpreter/JITc Changes—Long

- Single instruction loads and stores of longs

```
                        long x = y;
     G4

lwz     r3,y
lwz     r4,y+4
stw     r3,x
stw     r4,x+4
```

Interpreter/JITc Changes—Long

- Single instruction loads and stores of longs

```
long x = y;
```

G4

```
lwz     r3,y
lwz     r4,y+4
stw     r3,x
stw     r4,x+4
```

G5

```
ld      r3,y
std     r3,x
```

Interpreter/JITc Changes—Long

- Add, subtract, negate, and, or, xor (not)
  - Single instructions instead of two

Interpreter/JITc Changes—Long

- Add, subtract, negate, and, or, xor (not)
  - Single instructions instead of two

```
long z = x + y;
```

Interpreter/JITc Changes—Long

- Add, subtract, negate, and, or, xor (not)
  - Single instructions instead of two

```
                        long z = x + y;
```

G4

```
lwz     r3,x
lwz     r4,x+4
lwz     r5,y
lwz     r6,y+4
addc    r8,r4,r6
adde    r7,r3,r5
stw     r7,z
stw     r8,z+4
```

Interpreter/JITc Changes—Long

- Add, subtract, negate, and, or, xor (not)
  - Single instructions instead of two

```
long z = x + y;
```

G4

```
lwz     r3,x
lwz     r4,x+4
lwz     r5,y
lwz     r6,y+4
addc    r8,r4,r6
adde    r7,r3,r5
stw     r7,z
stw     r8,z+4
```

G5

```
ld      r3,x
ld      r4,y
add     r5,r3,r4
std     r5,x
```

Interpreter/JITc Changes—Long

- Multiply, divide, remainder, shifts, comparisons
  - Single instructions instead of many or runtime call

Interpreter/JITc Changes—Long

- Multiply, divide, remainder, shifts, comparisons
  - Single instructions instead of many or runtime call

```
long z = x * y;
```

Interpreter/JITc Changes—Long

- Multiply, divide, remainder, shifts, comparisons
  - Single instructions instead of many or runtime call

```
                    long z = x * y;
      G4

lwz     r3,x
lwz     r4,x+4
lwz     r5,y
lwz     r6,y+4
mullw   r9,r3,r6
mulhwu  r10,r4,r6
add     r9,r9,r10
mullw   r10,r4,r5
mullw   r7,r4,r6
add     r8,r9,r10
stw     r7,z
stw     r8,z+4
```

Interpreter/JITc Changes—Long

- Multiply, divide, remainder, shifts, comparisons
  - Single instructions instead of many or runtime call

```
long z = x * y;
```

G4

```
lwz     r3,x
lwz     r4,x+4
lwz     r5,y
lwz     r6,y+4
mullw   r9,r3,r6
mulhwu  r10,r4,r6
add     r9,r9,r10
mullw   r10,r4,r5
mullw   r7,r4,r6
add     r8,r9,r10
stw     r7,z
stw     r8,z+4
```

G5

```
ld      r3,x
ld      r4,y
mulld   r5,r3,r4
std     r5,x
```

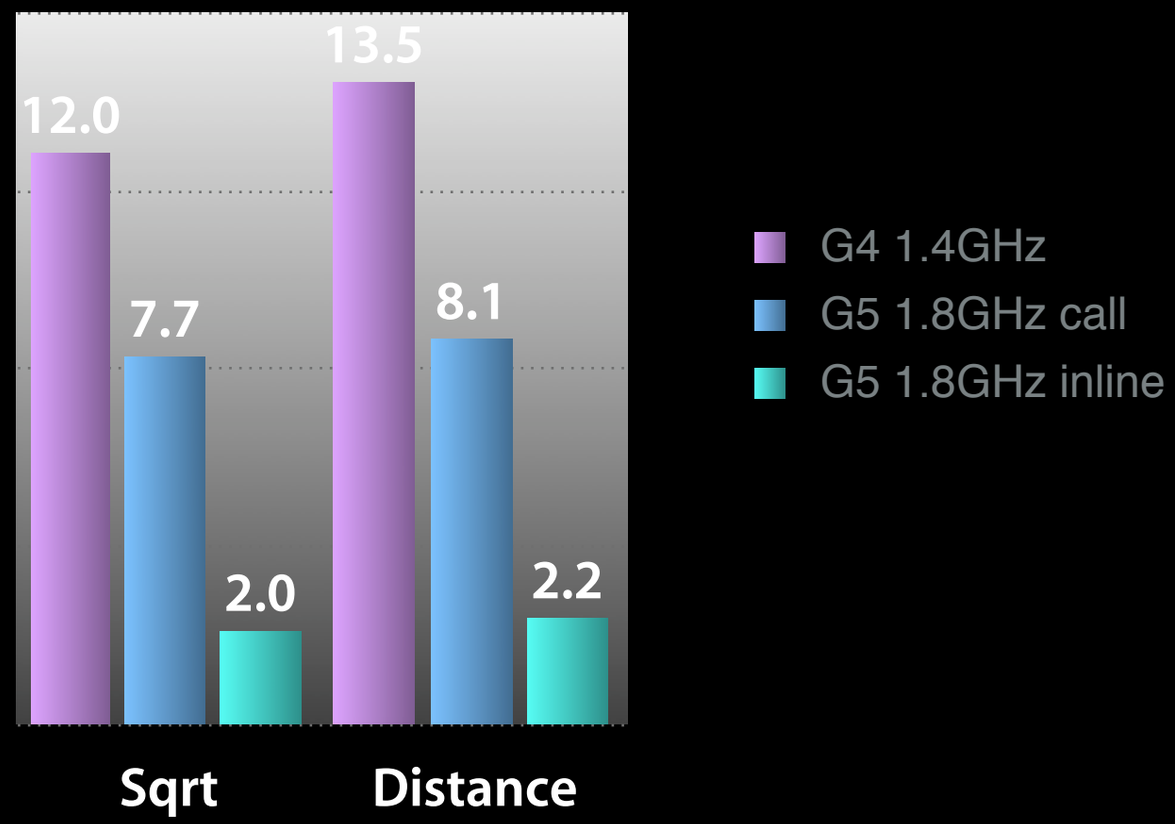
Interpreter/JITc Changed—Float

- Casts to/from long are now done inline
- Float/Double bit extractions (eg. doubleToLongBits) are now done inline
- Square root

Interpreter/JITc Changed—Float

- Casts to/from long are now done inline
- Float/Double bit extractions (eg. doubleToLongBits) are now done inline
- Square root

Library call versus inline fsqrt
100,000,000 data points
(Time in seconds)



Runtime Changes

- Volatile loads and stores
  - Using the `volatile` keyword on a field or static
  - Requires coordination of the low and high 32-bits
  - On the G4 we did a shuffle through a double register
  - On the G5 single instruction load or store

Runtime Changes

- Thread synchronization for memory allocation and monitors on dual processor machines use a new lightweight sync instruction (`lwsync`)
  - Doesn't require full disruption of the hardware state

Runtime Changes

- Atomic compare and exchange
  - sun.misc.AtomicLongCSImpl.attemptUpdate
  - Primarily used is in net operations
  - Multiprocessor access of long values
  - G4 uses Java synchronization
  - G5 uses lightweight load and reserve sequence

Summary

- Java 1.4.1 automatically adapts to the G5
- Significant performance wins across the board
  - Especially for long and floating point
- No java code changes required

Maximizing Java Performance

Optimal Java for Mac OS X

Victor Hernandez

Java Runtime Technologies

vhernandez@apple.com

Introduction

- HotSpot optimizes your application
  - Aggressive inlining in 1.4.1
- You can improve performance yourself
  - New I/O (NIO) in JDK 1.4.1
- Conclusions for optimizing heavily-used methods

Optimizing Java 1.4.1

- Situation:
  - A large cost of Java execution is the overhead of method invocation

- Approach:
  - Inline method calls during dynamic compilation

JIT Compilation: Inlining

- What is inlining?

```
int average(int a, int b) {
   int s = sum(a, b);
   return s / 2;
}

int sum(int x, int y) {
   return x + y;
}
```

```
int average(int a, int b) {
   int s = a + b;
   return s / 2;
}
```

JIT Compilation: Inlining

- What is inlining?

```
int average(int a, int b) {
   int s = sum(a, b);
   return s / 2;
}

int sum(int x, int y) {
   return x + y;
}
```

```
int average(int a, int b) {
   int s = a + b;
   return s / 2;
}
```

- 1.3.1 inlining:
  - accessors/new/intrinsics (sin, cos, ...)
  - no virtual methods

JIT Compilation: Inlining

- What is inlining?

```
int average(int a, int b) {
   int s = sum(a, b);
   return s / 2;
}

int sum(int x, int y) {
   return x + y;
}
```

```
int average(int a, int b) {
   int s = a + b;
   return s / 2;
}
```

- 1.3.1 inlining:
  - accessors/new/intrinsics (sin, cos, ...)
  - no virtual methods
- How could we inline your virtual methods?

JIT Compilation: Aggressive Inlining

- Class Hierarchy Analysis (CHA):
  - monomorphic: Only one implementation of a virtual method has been loaded
- HotSpot 1.4.1 aggressively inlines monomorphic methods
- Benefits:
  - Virtual methods can be inlined
  - If not inlined, virtual method invocations avoid virtual table lookup
  - Faster implementation of `instanceof` and `checkcast`

Optimizing in Java 1.4.1

- Situation:
  - Large amount of native data structures in Java Heap
  - JNI transitions required to perform System operations (graphics, networking, etc.)

- Approach:
  - Remove JNI dependency by providing direct access to native memory

New I/O: Native Buffers

- Native buffers
  - java.nio.* Buffer
  - Containers for data of primitive types
- I/O operations on java.nio.ByteBuffer
  - Directly-backed buffers

```
byte abyte = 0;
ByteBuffer buf = ByteBuffer.allocateDirect(400);
for (int i=0; i<400; i++) { buf.put(abyte); }

byte abyte = 0;
MappedByteBuffer buf = MappedByteBuffer.allocateDirect(400);
for (int i=0; i<400; i++) { buf.put(abyte); }
```

New I/O: Channels

- Channels
  - java.nio.channels.* Channel
  - Allows non-blocking and interruptible operations
- Improved File System support
  - java.nio.channels.FileChannel
    - File locking
    - Memory-mapped files

New I/O: Non-blocking I/O

- java.nio.channels.SocketChannel

```
public void initializeOperations()
  throws IOException, UnknownHostException
{
    sel    = Selector.open();
    server = ServerSocketChannel.open();
    server.configureBlocking(false);
    InetSocketAddress addr =
     new InetSocketAddress(InetAddress.getLocalHost(),
                           port);
    server.socket().bind(addr);
}
```

New I/O: Non-blocking I/O

- SocketChannel Selector Model

```
public void startServer() throws IOException {
    initializeOperations();

  SelectionKey acceptKey = server.register(sel, SelectionKey.OP_ACCEPT);
    while (acceptKey.selector().select() > 0 ) {
      Iterator it = sel.selectedKeys().iterator();
        while (it.hasNext()) {
          SelectionKey key = (SelectionKey)it.next();
          it.remove();
          if (key.isAcceptable()) {
            ServerSocketChannel ssc = (ServerSocketChannel) key.channel();
            socket = (SocketChannel) ssc.accept();
            socket.configureBlocking(false);
            SelectionKey another =
              socket.register(sel, SelectionKey.OP_READ | SelectionKey.OP_WRITE);
          }
          if (key.isReadable()) { ... }
          if (key.isWritable()) { ... }
```

New I/O: Trade-offs

- New I/O is not free
  - Allocation/deallocation costs differ from Java arrays
- `get`/`put` methods of Native Buffers are not all inlined

New I/O: Trade-offs

- New I/O is not free
  - Allocation/deallocation costs differ from Java arrays
- `get`/`put` methods of Native Buffers are not all inlined
- Cost is balanced when:
  - Heavy use of System APIs with native data structures
  - Reducing Thread overhead in Server I/O

How to write Optimal Java

- Maximize inlining opportunities
  - Write small concise methods
  - Use accessor methods
  - `final` is superficial for performance tuning
  - Most JDK methods can be inlined

  - Unable to inline:
    - `synchronized`
    - Large methods
    - Methods with exception handlers

How to write Optimal Java

- Avoid object pools
  - new is inlined
  - Thread-Local allocation reduces contention
  - Precise garbage collection

How to write Optimal Java

- Avoid object pools
  - new is inlined
  - Thread-Local allocation reduces contention
  - Precise garbage collection

- Avoid programming by exception
  - Creating exception objects are expensive
  - Can prevent inlining
  - Cost only when exception is thrown

Maximizing Java Performance

Graphics

Gerard Ziemski
Java Runtime Classes
gziemski@apple.com

Introduction

- A review of Java graphics on Mac OS X
- Tips and techniques on hand tuning graphics performance of your Java applications for Mac OS X
- Demo

Past—Java Graphics on Mac OS X

- Java 1.3.1
  - Java2D hardware acceleration

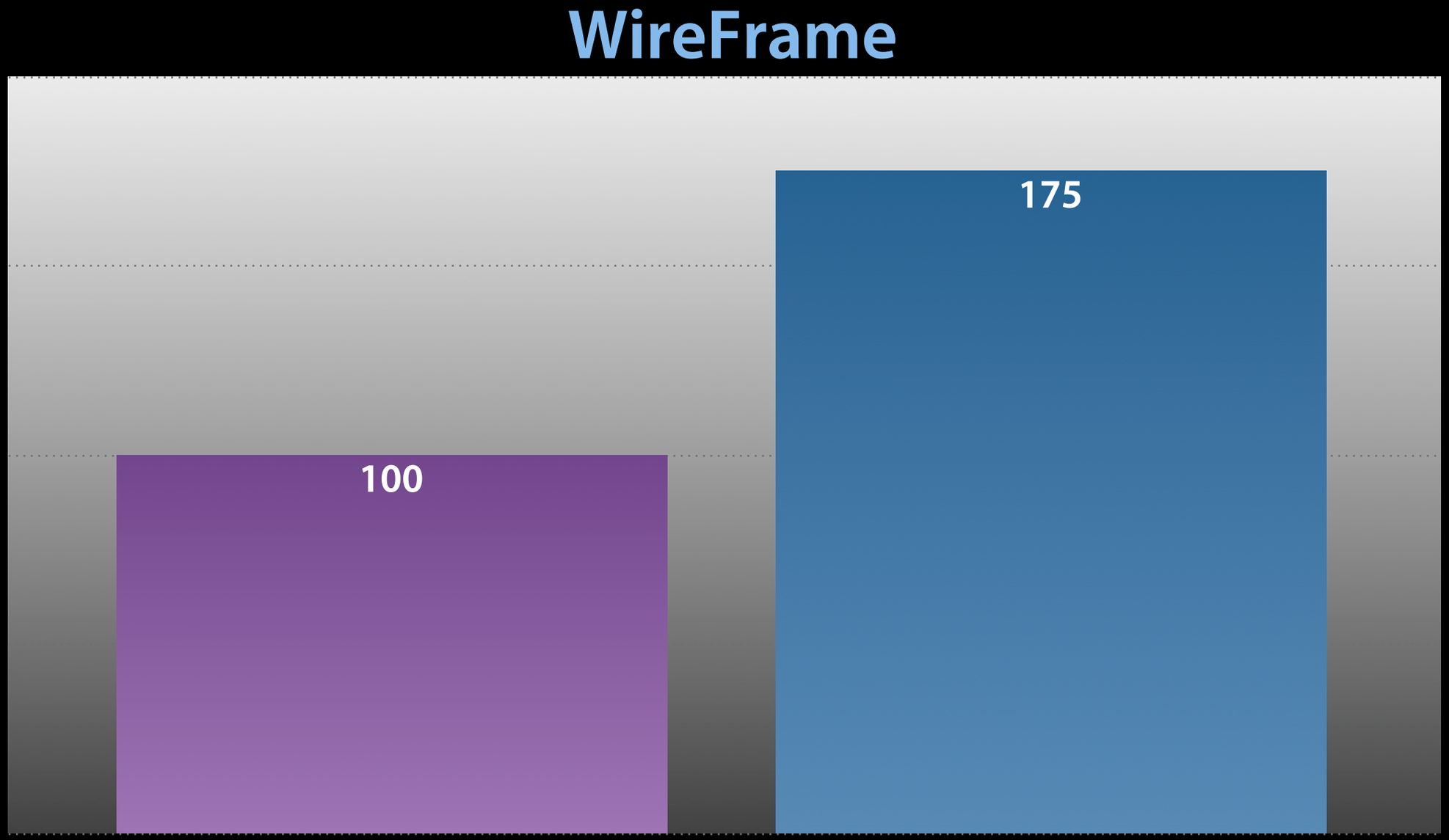
- Java 1.4.1
  - Rearchitecture from Carbon to Cocoa

Java 1.4.1 Update 1 on Mac OS X

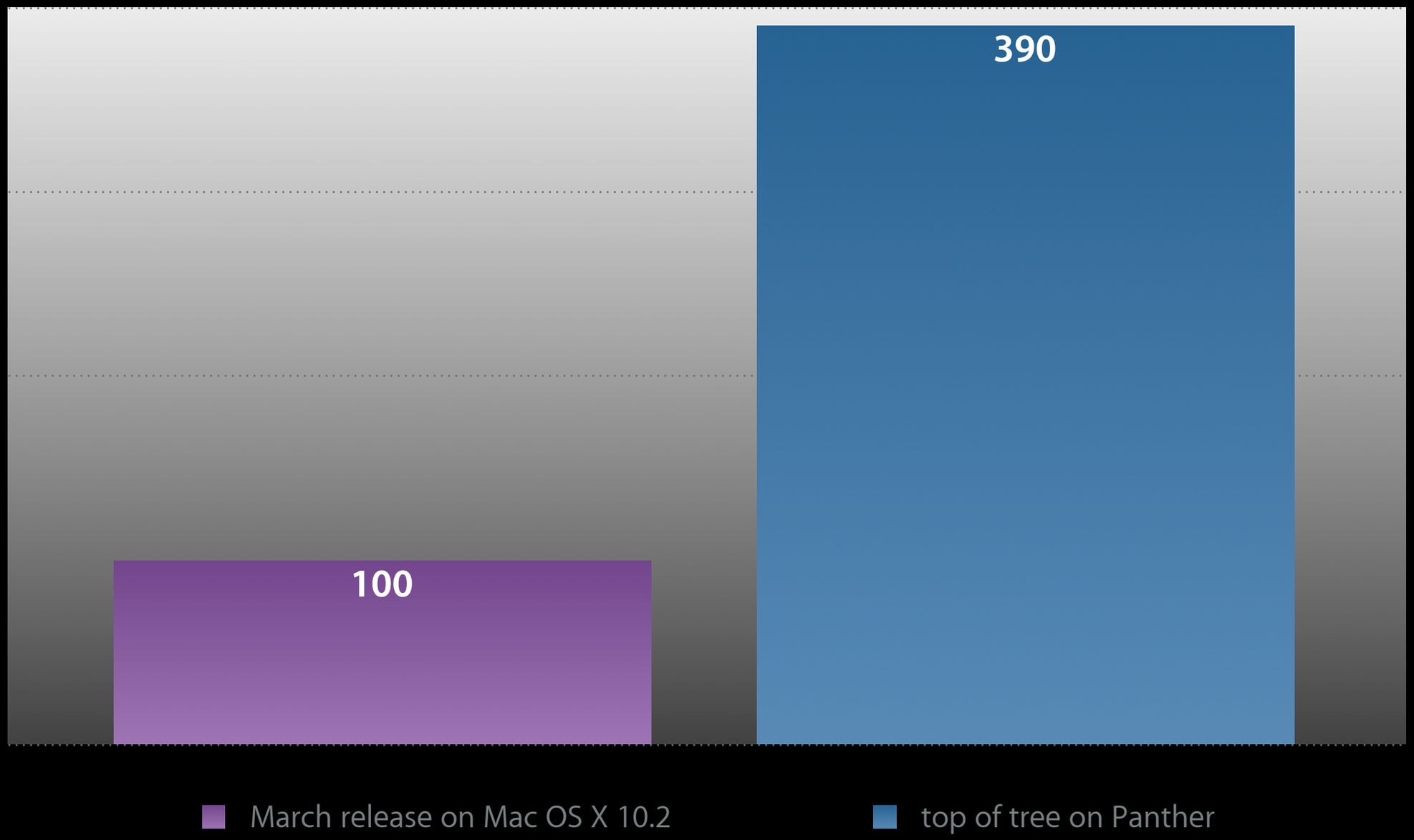
- Correctness

- Optimizations
  - Lazy drawing
  - Lazy pixel conversion
  - Lazy state management

Java 1.4.1 Graphics Benchmarks



Java 1.4.1 Graphics Benchmarks



Continuing Work

- Lazy state management
- More optimizations
- Collaboration with Sun Java2D Engineering
- Tighter integration with CoreGraphics

Image Formats

- Optimizations tips and techniques

Image Format Tips and Techniques

- Avoid hardcoding BufferedImage format types

```
BufferedImage bim = new BufferedImage(w, h, 2);

Graphics g = bim.createGraphics();
g.setColor(Color.red);
g.fillRect(0, 0, w, h);
g.dispose();
```

Image Format Tips and Techniques

- Use createCompatibleVoltileImage to create a BufferedImage

```
GraphicsEnvironment ge =
GraphicsEnvironment.getLocal GraphicsEnvironment();
GraphicsConfiguration gc =
ge.getDefaultConfiguration();
BufferedImage bim =
gc.createCompatibleVolatileImage();

Graphics g = bim.createGraphics();
g.setColor(Color.red);
g.fillRect(0, 0, w, h);
g.dispose();
```

Image Format Tips and Techniques

- In Java on Mac OS X, indexed color formats use more memory than other image formats

- The most optimal image format might vary depending on capabilities of your computer and the screen depth of your monitor

Image Format Tips and Techniques

- Natively supported image types as destination:
  - BufferedImage.TYPE_INT_RGB
  - BufferedImage.TYPE_INT_ARGB_PRE
  - BufferedImage.TYPE_USHORT_555_RGB
  - BufferedImage.TYPE_BYTE_GRAY

Image Format Tips and Techniques

- Natively supported image types as source:
  - BufferedImage.TYPE_INT_RGB
  - BufferedImage.TYPE_INT_ARGB
  - BufferedImage.TYPE_INT_ARGB_PRE
  - BufferedImage.TYPE_USHORT_555_RGB
  - BufferedImage.TYPE_BYTE_GRAY

Rendering

- Optimizations tips and techniques

Rendering Tips and Techniques

- If you need to render on a thread different than AWT event dispatch thread:
  - First, render into an image
  - Second, draw that image to the screen

Rendering Tips and Techniques

- The fastest way to draw a pixel into a natively supported image, is to write directly to the image's underlaying pixel array backing store

```
BufferedImage bim;

WritableRaster wr = bim.getRaster();
DataBuffer db = wr.getDataBuffer();
int[] pixels = db.getData();

pixels[0] = 0xff0000ff;
```

Rendering Tips and Techniques

- The fastest way to draw a pixel into a non–natively supported image, is to create a Graphics object and use it to draw into the image

```
BufferedImage bim;

Graphics g = bim.createGraphics();
g.setColor(Color.blue);
g.fillRect(0, 0, 1, 1);
```

Other

- Optimizations tips and techniques

Other Tips and Techniques

- Avoid creating new objects in your paint method
- Use simple primitives instead of shapes
- Use polyline instead of drawing lines one at a time
- Use bytes, not chars for text drawing
- Use double buffering for rendering complex scenes
- Use RuntimeOptions to help you understand your application's behavior

Demo

jogl

Ken Russell
Sun Microsystems

Wrap Up

Desktop Java Roadmap

608 Creating Great Java Applications on Mac OS X	Nob Hill Wed., 9:00am
619 Java and the Web	Nob Hill Thurs., 10:30am
620 Java Native Integration on Mac OS X	Nob Hill Thurs., 2:00pm
625 Maximizing Java Performance on Mac OS X	Russian Hill Fri., 3:30pm
708 QuickTime APIs Update	Pacific Heights Tues., 5:00pm
FF014 Feedback Session	The Marina Fri., 10:30am

Java Contacts

Alan Samuel

Java Evangelist

bluker1@apple.com

Bob Fraser

Java Product Marketing Manager

bfraser@apple.com

Java Developer Mailing List

Java-dev

http://lists.apple.com

Reference Library

Documentation

Java Developer Documentation

http://developer.apple.com/documentation/java

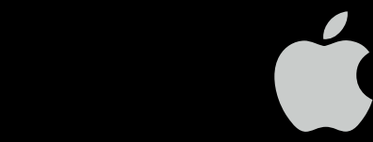
Java 1.4.1 Virtual Machine System Properties

http://developer.apple.com/documentation/java

Documentation (WWDC Private)

Java 1.4.1 System Properties

ADC Member Site > Download Software > WWDC 2003 > Mac OS X "Panther" and Xcode Prerelease Documentation



Reference Library

Other Resources

ADC Java Developer Site

http://developer.apple.com/java

Sun's Java Web Site

http://java.sun.com

Java 1.4.1 Update 1 Developer Preview

http://connect.apple.com

QuickTime for Java Seed

http://connect.apple.com

WWDC Uber Link List

http://developer.apple.com/wwdc2003/url.html

Q&A

Victor Hernandez, Jim Laskey, and Gerard Ziemski

Java Runtime Technologies and Runtime Classes

vhernandez, jlaskey, gziemski@apple.com

http://developer.apple.com/wwdc2003/urls.html

WWDC 2003

Worldwide Developers Conference